arXiv:cs/0610059v2 [cs.CV] 27 Mar 2008

Camera motion estimation through planar deformation determination

C. Jonchery[(1)], F. Dibos[(2)] and G. Koepfler[(1)]

March 28, 2008

[(1)] MAP5 Université Paris 5,
45, rue des Saints-Pères 75270 Paris Cedex 06, FRANCE
claire.jonchery@math-info.univ-paris5.fr
georges.koepfler@math-info.univ-paris5.fr

[(2)] LAGA, L2TI Université Paris 13
99, avenue Jean-Baptiste Clment 93430 Villetaneuse, FRANCE
email: dibos@math.univ-paris13.fr

Abstract

In this paper, we propose a global method for estimating the motion of a camera which films a static scene. Our approach is direct, fast and robust, and deals with adjacent frames of a sequence. It is based on a quadratic approximation of the deformation between two images, in the case of a scene with constant depth in the camera coordinate system. This condition is very restrictive but we show that provided translation and depth inverse variations are small enough, the error on optical flow involved by the approximation of depths by a constant is small. In this context, we propose a new model of camera motion, that allows to separate the image deformation in a similarity and a "purely" projective application, due to change of optical axis direction. This model leads to a quadratic approximation of image deformation that we estimate with an M-estimator; we can immediatly deduce camera motion parameters.

1 Introduction

The estimation of camera motion plays a crucial role in many domains of computer vision such as the recovery of scene structure, medical imaging, augmented reality and so on. This is a difficult task since the motion of a pixel between two images depends not only on the six parameters of camera motion between the two successive image captures, but also on the depth at the corresponding point in the static scene. Existing methods can be classified as features correspondences-based approaches, which are local, optical flow methods and direct methods, which are global.

Among all proposed methods using features correspondences, one can mention recursive techniques based on extended Kalman filters [1, 2] which track camera motion and estimate the structure of the scene. The essential matrix, which was first defined by Longuet-Higgins in [3], is often estimated, as only a few correspondences in two images are sufficient; the number of required correspondences is discussed by Faugeras et al. in [4, 5, 6]. In the case of an uncalibrated camera, the analogous approach is described in [7] with the fundamental matrix.

The use of optical flow avoids the choice of "good" features; many authors use the basic bilinear constraint linking optical flow, camera velocities and depths of projected points; in [8], Bruss and Horn apply an algebraic computation to remove depth from the bilinear constraint and use numerical optimization techniques. Heeger and Jepson, in [9], decouple the translational velocity from the rotational velocity and use linear subspace methods. Ma et al. in [10] and Brooks et al. in [11] use a different approach with the epipolar differential constraint: a differential essential matrix is determined from the optical flow, leading to a unique camera velocity estimation. Another well-known approach is based on motion parallax, notably developped by Tomasi and Shi in [12], Lawn and Cipolla in [13] and Irani et al. in [15]. Tomasi et al. propose in [14] a comparison of algorithms which only use optical flow for estimating camera motion.

Finally, direct methods use directly the content of a couple of images. They are generally based on the constraint of constant illumination (also called optical flow constraint), that is minimized by a least square approach, on the parameters of a given motion model. Different assumptions are used to avoid estimating depths on all points; for example, Horn and Weldon in [16] and Bergen et al., in [17], assume that the depth map is locally constant. In [18], Negahdaripour and Horn consider that it is planar or quadratic.

Let us notice that features correspondences-based techniques work best with well separated views, when the displacement (especially the translation or the so-called baseline) between frames is sufficiently large. On the contrary, optical flow methods and direct methods, based on infinitesimal approximations, are well-adapted to very small motions.

Our method deals with adjacent frames of a sequence, so with narrow baselines and restricted camera rotations. It is a direct method, very fast and robust, based on a quadratic approximation of image deformation.

The outline of the paper is as follows. In Section 2, we describe our framework. We recall the image deformation generated by camera motion. Then, we show that we can assume in the deformation formula that depth of projected points is constant (in camera coordinate system) under following condition: the product of the norm of translation with the maximal variation of inverse depth has to be sufficiently small. Thus, two consecutive images are linked by a planar transformation. In this context, we introduce in Section 3 the registration group, used for modeling image deformation generated by a camera displacement. We also propose a new camera motion decomposition, that separates image deformation in a "purely" projective deformation, due to change of optical axis direction, and a similarity. As camera displacement is restricted, we obtain a quadratic approximation of optical flow between two adjacent frames. This approximation is used in Section 4 to define an algorithm of motion estimation; we show estimation results on synthetic sequences and use motion estimations on real video sequences for mosaicing and simplified augmented reality. Concluding remarks are given in Section 5.

2 Framework

2.1 Pinhole camera model

A camera projects a point in 3D space on a 2D image. This transformation can be described using the well-known pinhole camera model [7] presented in figure 1. The camera is located on C, the optical center, and directed by k, the optical axis. The camera projects a point M of the 3D space on the plane $\mathcal{R} : \{Z = f_c\}$. The plane $\mathcal{R}$ is called the retinal plane and f_c the focal length. The projection m of M is then the intersection of the optical ray (CM) with $\mathcal{R}$.

Let c be the intersection of the optical axis with $\mathcal{R}$. If (X, Y, Z) are the coordinates of M in the camera coordinate system (C, i, j, k) and (x, y) the coordinates of m in the orthogonal basis (c, i, j), the relationship between (x, y) and (X, Y, Z) is following

$$\begin{cases} x = f_c \frac{X}{Z} \\ \\ y = f_c \frac{Y}{Z}. \end{cases}$$

As f_c just acts as a scaling factor on the image, we choose in this paper, without loss of generality, to set the focal length to one. Then, f_c will be the unit of camera and image coordinate systems.

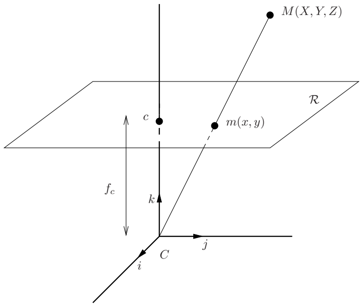

Figure 1: *Pinhole camera model.*

2.2 Camera motion

Let D be a displacement of the camera or in an equivalent way a displacement of the plane $\mathcal{R}$. The movement D may be written in a unique way as $D = (R, t)$, where R is a rotation with axis containing C and t a translation. The set of displacements $D = (R, t)$ forms the Lie group of rigid transformations in $\mathbb{R}^3$ called $SE(3)$, which denotes the special Euclidian group. The displacement $D = (R, t)$ transforms a point M belonging to $\mathbb{R}^3$ in $M' = RM + t$. Thus, the camera is identified before the displacement by (C, i, j, k) and after the displacement by $(C', R(i), R(j), R(k))$, with $CC' = t$. In the following, we denote

$$R = \begin{pmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{pmatrix} \qquad \text{and} \qquad t = \begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix}.$$

Let now f and g be two adjacent images in a sequence defined on rectangular domains K of $\mathcal{R}$ and K' of $\mathcal{R}'$ (with $f_c = 1$). Let M be a point in $\mathbb{R}^3$ such that its projections m and m' on $\mathcal{R}$ and $\mathcal{R}'$ belong to K and K'. We denote $m = (x, y)$ in (c, i, j) and $m' = (x', y')$ in $(c', R(i), R(j))$. Thus, if we make the assumption of constant illumination, we have

$$f(x, y) = g(x', y'),$$

and the two points are linked by

$$\begin{cases} x' = \dfrac{a_1 x + a_2 y + a_3 - \langle \frac{t}{Z(x,y)}, R(i) \rangle}{c_1 x + c_2 y + c_3 - \langle \frac{t}{Z(x,y)}, R(k) \rangle} \\ \\ y' = \dfrac{b_1 x + b_2 y + b_3 - \langle \frac{t}{Z(x,y)}, R(j) \rangle}{c_1 x + c_2 y + c_3 - \langle \frac{t}{Z(x,y)}, R(k) \rangle} \end{cases} \tag{1}$$

and

$$\begin{cases} x = \dfrac{a_1 x' + b_1 y' + c_1 + \frac{t_1}{Z'(x',y')}}{a_3 x' + b_3 y' + c_3 + \frac{t_3}{Z'(x',y')}} \\ \\ y = \dfrac{a_2 x' + b_2 y' + c_2 + \frac{t_2}{Z'(x',y')}}{a_3 x' + b_3 y' + c_3 + \frac{t_3}{Z'(x',y')}}, \end{cases} \tag{2}$$

where $Z(x, y)$ and $Z'(x', y')$ are the depths of M respectively in (C, i, j, k) and (C', $R(i)$, $R(j)$, $R(k)$).

2.3 Depths approximation by a constant

We now wish to approximate the depths by a constant in the two formulas (1) and (2). Let Z_0 belong to $\mathbb{R}_+^*$. By a Taylor expansion of equation (1) on $\frac{1}{Z(x,y)}$ about $\frac{1}{Z_0}$, we obtain

$$\left\{\begin{array}{ll} x' = & \dfrac{a_1x + a_2y + a_3 - \langle \frac{t}{Z_0}, R(i)\rangle}{c_1x + c_2y + c_3 - \langle \frac{t}{Z_0}, R(k)\rangle} + \\ & \left(\frac{1}{Z(x,y)} - \frac{1}{Z_0}\right)\left(-\langle t, R(i)\rangle + \langle t, R(k)\rangle \frac{a_1x+a_2y+a_3}{\left(c_1x+c_2y+c_3-\langle \frac{t}{Z_0}, R(k)\rangle\right)^2}\right) \\ & + o\left(\frac{1}{Z(x,y)} - \frac{1}{Z_0}\right) \\ & \\ y' = & \dfrac{b_1x + b_2y + b_3 - \langle \frac{t}{Z_0}, R(j)\rangle}{c_1x + c_2y + c_3 - \langle \frac{t}{Z_0}, R(k)\rangle} + \\ & \left(\frac{1}{Z(x,y)} - \frac{1}{Z_0}\right)\left(-\langle t, R(j)\rangle + \langle t, R(k)\rangle \frac{b_1x+b_2y+b_3}{\left(c_1x+c_2y+c_3-\langle \frac{t}{Z_0}, R(k)\rangle\right)^2}\right) \\ & + o\left(\frac{1}{Z(x,y)} - \frac{1}{Z_0}\right). \end{array}\right.$$

Thus, if for all $(x, y) \in K$, $\left(\frac{1}{Z(x,y)} - \frac{1}{Z_0}\right) \|t\|$ is small enough with respect to the image coordinates, we can substitute Z_0 in place of $Z(x, y)$.

We now make some numerical and technical assumptions that are little restrictive and so are likely verified by a couple of consecutive images.

Hypothesis 1 – *Let $D = (R, t) \in SE(3)$ and K be the rectangular domain where f is defined. Let Z be the depth function of projected points, defined on K. We assume that*

$$\left| \frac{1}{c_1x + c_2y + c_3 - \langle \frac{t}{Z(x,y)}, R(k)\rangle} \right| \leq \frac{4}{3}.$$

Hypothesis 2 – *Let $D = (R, t) \in SE(3)$ and K be the rectangular domain where f is defined, having maximal dimension L. Let Z be the depth function of projected points, defined on K. For two matching points (x, y) and (x', y') (in the sense of formulas (1) and (2)), we suppose that*

$$\max\{|x' - x|, |y' - y|\} \leq \frac{L}{2}.$$

The first hypothesis comes from the fact that the variation of optical axis direction and its translation along the axis k, between two consecutive acquisitions, have to be very small so that images were workable. The second one formulates the limitation of points displacements between two images; we assume that the two components of optical flow can not be larger than the half of image larger dimension.

With these two assumptions, we show in Appendix A the following theorem.

Theorem 1 – *Let $D = (R, t) \in SE(3)$ and K be the rectangular domain where f is defined, and having maximal dimension L. Let Z be the depth function of projected points, defined on K, bounded by $Z_{inf} > 0$ and Z_{sup}. We assume that Z and D verify hypothesis 1 and 2. If*

$$\left(\frac{1}{Z_{inf}} - \frac{1}{Z_{sup}}\right) \|t\| \frac{2\,(L+1)}{3} \leq \varepsilon \tag{3}$$

then there exists $Z_0 > 0$ so that we can replace $Z(x, y)$ by Z_0 in the equations (1) with an error bounded by ε.

The value of Z_0 that minimizes ε is

$$\widehat{Z}_0 = \arg\min_{Z_0} \max_{(x,y)\in K} \left| \frac{1}{Z(x,y)} - \frac{1}{Z_0} \right| = \frac{2Z_{sup}\,Z_{inf}}{Z_{sup}+Z_{inf}}.$$

We can also show that we can substitute the same Z_0 in place of $Z'(x',y')$ in equations (2) with an error bounded by $\varepsilon + \varepsilon'$ if

$$\frac{4}{9Z_{inf}}\, \|t\|\, (L+1) < \varepsilon'. \tag{4}$$

For small values of ε and ε', conditions (3) and (4) can be verified in the following cases:

- if there is no translation, depths do not appear in formulas (1) and (2),
- if $t \neq 0$, the scene must be far enough from the camera for verifying condition (4). The variations of amplitude of $1/Z$ must also be small enough for verifying condition (3): the further the scene takes place from the camera, the bigger are the authorized variations of depth.

With this framework, relations (1) and (2) between f and g become

$$f(x,y) = g\left(\frac{a_1x + a_2y + a_3 - \langle \widetilde{t}, R(i)\rangle}{c_1x + c_2y + c_3 - \langle \widetilde{t}, R(k)\rangle}, \frac{b_1x + b_2y + b_3 - \langle \widetilde{t}, R(j)\rangle}{c_1x + c_2y + c_3 - \langle \widetilde{t}, R(k)\rangle} \right) = g \circ \psi(x,y)$$

and

$$g(x',y') = f\left(\frac{a_1x' + b_1y' + c_1 + \widetilde{t}_1}{a_3x' + b_3y' + c_3 + \widetilde{t}_3}, \frac{a_2x' + b_2y' + c_2 + \widetilde{t}_2}{a_3x' + b_3y' + c_3 + \widetilde{t}_3} \right) = f \circ \varphi(x',y'),$$

where $\widetilde{t} = \frac{t}{Z_0}$. In the sequel of the paper, we will assume that conditions (3) and (4) are verified: we will use applications φ and ψ as the relations between f and g. As we will consider two consecutive images in a sequence, the translation t is very small.

3 Modelisation

We now consider two consecutive images f and g in a sequence, obtained before and after a camera motion $D = (R, t)$.

3.1 Registration group

The applications φ and ψ are projective applications, each defined by six parameters, three for the rotation and three for the translation. Projective applications are classically represented in the projective group in $\mathbb{R}^2$. This group is isomorphic to the special linear group $SL(\mathbb{R}^3)$ of invertible matrices. Thus, the applications φ and ψ are associated to the following invertible matrices $\mathcal{M}_\varphi$ and $\mathcal{M}_\psi$

$$\mathcal{M}_\varphi = \begin{pmatrix} a_1 & b_1 & c_1 + \widetilde{t}_1 \\ a_2 & b_2 & c_2 + \widetilde{t}_2 \\ a_3 & b_3 & c_3 + \widetilde{t}_3 \end{pmatrix} = R \begin{pmatrix} 1 & 0 & \langle \widetilde{t}, R(i)\rangle \\ 0 & 1 & \langle \widetilde{t}, R(j)\rangle \\ 0 & 0 & 1 + \langle \widetilde{t}, R(k)\rangle \end{pmatrix} = RH \tag{5}$$

and

$$\mathcal{M}_\psi = \begin{pmatrix} a_1 & a_2 & a_3 - \langle \widetilde{t}, R(i)\rangle \\ b_1 & b_2 & b_3 - \langle \widetilde{t}, R(j)\rangle \\ c_1 & c_2 & c_3 - \langle \widetilde{t}, R(k)\rangle \end{pmatrix} = R^{-1} \begin{pmatrix} 1 & 0 & -\widetilde{t}_1 \\ 0 & 1 & -\widetilde{t}_2 \\ 0 & 0 & 1 - \widetilde{t}_3 \end{pmatrix} = R^{-1}\widetilde{H}.$$

Our aim is to estimate camera motion through image deformation, each defined by six parameters. But the projective group is an eight parameters group and the matrix decomposition shows that $\mathcal{M}_\varphi^{-1} \neq \mathcal{M}_\psi$ in $SL(\mathbb{R}^3)$. Thus we are going to model the projective transformation in another group, well-adapted: the registration group, introduced by Dibos in [19].

Definition 1 – *Let $\mathcal{A}$ be the subset of projective applications*

$$\mathcal{A} = \Big\{ \ \phi : \mathbb{R}^2 \to \mathbb{R}^2 \textit{ so that } \forall (x,y) \in \mathbb{R}^2,$$

$$\phi(x,y) = \left(\frac{a_1 x + b_1 y + c_1 + \alpha}{a_3 x + b_3 y + c_3 + \gamma}, \frac{a_2 x + b_2 y + c_2 + \beta}{a_3 x + b_3 y + c_3 + \gamma} \right),$$

$$\textit{where } R = \begin{pmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{pmatrix} \in SO(3) \ \textit{ and } \ (\alpha, \beta, \gamma) \in \mathbb{R}^3 \ \Big\}.$$

The registration group is $(\mathcal{A}, \star)$, where the composition law $\star$ is deduced from the composition law $\circ$ of $SE(3)$ through the isomorphism

$$\mathcal{I} : \mathcal{A} \longrightarrow SE(3)$$

$$\forall \phi \in \mathcal{A} \quad \mathcal{I}(\phi) = (R, t)$$

where R is the rotation defined above and $t = (\alpha, \beta, \gamma)$ is the translation.

More precisely, let ϕ_1 and ϕ_2 belong to $\mathcal{A}$, they correspond to the displacements $D_1 = (R_1, t_1)$ and $D_2 = (R_2, t_2)$, respectively. Then, $\phi_1 \star \phi_2 = \phi$ where ϕ is the projective application associated to the displacement $D = D_1 \circ D_2 = (R, t)$ where t is the translation with vector $t = t_1 + R_1 \, t_2$ and $R = R_1 R_2$. The notation $D_1 \circ D_2$ means that the camera first performs the displacement D_1 and second D_2. Moreover, if ϕ belongs to $\mathcal{A}$ and is associated to $D = (R, t)$, then ϕ^{-1} is associated to $D^{-1} = (R^{-1}, -R^{-1} t)$.

The applications φ and ψ belong to $\mathcal{A}$; we have $g(x,y) = f(\varphi(x,y))$ and $f(x,y) = g(\psi(x,y))$ with $\psi = \varphi^{-1}$ in the registration group (but not in the projective group).

By modeling the camera displacement in the registration group, we reduce the problem to the determination of six parameters of a planar application, as R and t are respectively defined by three parameters.

3.2 Camera motion decomposition

We propose here to decompose a camera motion in order to separate the image deformation in two components: a similarity part and a “purely” projective part. Indeed, any camera motion can be decomposed into three basic types of motion:

- a translation, which produces an homothety translation on the image f belonging to the plane $\mathcal{R}$,
- a rotation with axis k, which produces a planar rotation on f,
- a rotation with axis in the plane (C, i, j) which distorts f.

3.2.1 Decomposition of rotation

Let us consider a camera rotation R with axis containing C. We decompose R in two particular rotations $R_2 R_1$. The first one R_1, with axis Δ belonging to the plane (C, i, j) transforms the direction of the optical axis k in $R(k)$; this rotation induces a projective deformation of the image f. The second one R_2 is a rotation with axis $R(k)$: R_2 induces a planar rotation of the image $R_1(f)$. Any camera rotation can be written in such a way.

This decomposition is interesting because of the induced deformations of the image. R_1 produces a "purely" projective deformation of the image f whereas R_2 creates a planar rotation of the image $R_1(f)$.

Let us express the rotation R_1 with two parameters: θ for the location of Δ in the plane (C, i, j) and α for the angle of the rotation. If we denote R_a^l the rotation matrix with axis l and angle a, the expression of R_1 in (C, i, j, k) is

$$R_1 = R_\theta^k R_\alpha^i R_{-\theta}^k$$

which we denote in the following $R_{\theta,\alpha}$. Now, let β be the angle of the rotation R_2 around the new optical axis $R(k)$. We can then write the rotation R_2 in (C, i, j, k)

$$R_2 = R_\theta^k R_\alpha^i R_\beta^k R_{-\alpha}^i R_{-\theta}^k.$$

Finally, the expression of the global rotation R is

$$R = R_2 R_1 = R_\theta^k R_\alpha^i R_\beta^k R_{-\theta}^k = R_{\theta,\alpha} R_\beta^k.$$

Thus, the rotation R may also be decomposed in a rotation around the axis k followed by the rotation $R_{\theta,\alpha}$.

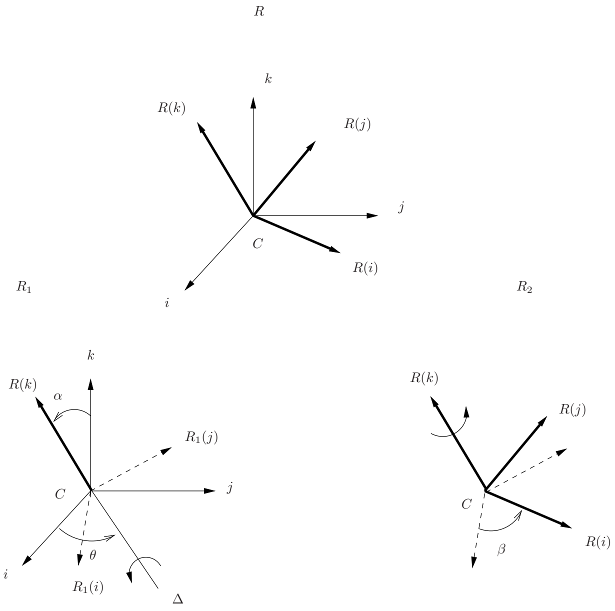

Figure 2: *Decomposition of a camera rotation R in two rotations $R_2 R_1$.*

3.2.2 Decomposition of a complete motion

A complete camera motion $D = (R, t)$ induces a projective deformation φ of the image f. The matrix associated to φ is RH, according to formula (5), which can now be written as

$$RH = R_{\theta,\alpha} R_\beta^k H.$$

If we denote $r_{\theta,\alpha}$ the "purely" projective deformation associated to the rotation $R_{\theta,\alpha}$ and s the similarity associated to $R_\beta^k H$ then we have

$$g(x, y) = f(\varphi(x, y)) = f(r_{\theta,\alpha} \circ s(x, y)) = f \circ r_{\theta,\alpha} \circ s(x, y).$$

We obtain therefore six parameters defining the camera motion, two for the rotation $R_{\theta,\alpha}$ and four for the translation t and rotation R^k_β. We express now camera motion with the following parameters $(\theta, \alpha, \beta, A, B, C)$ where $(-A, -B, -C)$ are the coordinates of t in the basis $(R(i), R(j), R(k))$. These new notations allow to obtain an easier writting of the projective application ψ (the inverse of φ in the registration group), which we will use later

$$\psi(x, y) = \left(\frac{a_1 x + a_2 y + a_3 + A}{c_1 x + c_2 y + c_3 + C}, \frac{b_1 x + b_2 y + b_3 + B}{c_1 x + c_2 y + c_3 + C} \right). \tag{6}$$

Remark that the six parameters $(\theta, \alpha, \beta, A, B, C)$ allow to access explicitly the camera displacement $D = (R, t)$. Indeed,

$$\begin{cases} \widetilde{t} = -AR(i) - BR(j) - CR(k) \\ \\ R = R_{\theta,\alpha} R^k_\beta. \end{cases}$$

3.3 Parameter values

As we consider two successive images of a video sequence with a high frame rate (classically 24 images per second), the camera motion between two images is very small and the parameter values are restricted, except for the angle θ which belongs to $]-\pi, \pi]$. Let us remark that the dimensions of K and K' verify a practical constraint: the view angle of a camera is usually not larger than $150°$. This means that L, the maximal dimension of K, must verify $L \leq 8\, f_c$, as the relation between the view angle a, f_c and L, illustrated on figure 3, is

$$\tan \frac{a}{2} = \frac{L}{2\, f_c}.$$

As $f_c = 1$, we have $L \leq 8$.

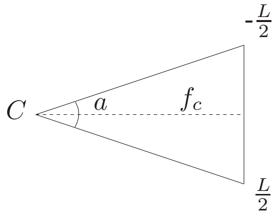

Figure 3: *Relation between the view angle a of the camera, the focal length f_c and the maximal dimension L of images.*

Table 1 gives orders of magnitude of parameter values that we have obtained by experiment, when we take a unit focal length. These experiments consist in taking images and applying the six parameters projective application. As the images have not to be too deformed, we deduce the orders of magnitude of parameters.

Parameter	Values
θ (radian)	$]-\pi, \pi]$
α (radian)	$[0, 0.03]$
β (radian)	$[-0.05, 0.05]$
A,B	$[-0.09, 0.09]$
C	$[-0.03, 0.03]$

Table 1: *Parameter values (A, B and C are expressed in units of focal length).*

3.4 Optical flow approximation

Theorem 2 – *Let us consider a scene orthogonal to the axis k. Let $D = (R, t)$ belong to $SE(3)$, also denoted $D = (\theta, \alpha, \beta, A, B, C)$. Let K and K' be the domains where f and g are defined, with maximal dimension L, and (x, y) and (x', y') two matching points of K and K'. We assume that hypothesis 1 is verified, $|\alpha| < 1$ and $|\beta| < 1$. Then, the optical flow at (x, y) verifies*

$$\left\{\begin{array}{ll} x' - x = & -Cx + A + \beta y + \alpha x(y\cos\theta - x\sin\theta) - \alpha\sin\theta + o(C) + o(\alpha) + o(\beta) \\ & +o(\sqrt{|\alpha A|}) + o(\sqrt{|\alpha C|}) + o(\sqrt{|AC|}) + o(\sqrt{|C\beta|}) + o(\sqrt{|\alpha\beta|}) \\ & \\ y' - y = & -Cy + B - \beta x + \alpha y(y\cos\theta - x\sin\theta) + \alpha\cos\theta + o(C) + o(\alpha) + o(\beta) \\ & +o(\sqrt{|\alpha B|}) + o(\sqrt{|\alpha C|}) + o(\sqrt{|BC|}) + o(\sqrt{|C\beta|}) + o(\sqrt{|\alpha\beta|}) \end{array}\right.$$

and

$$\left\{\begin{array}{l} \left|x' - x - (-Cx + A + \beta y + \alpha x(y\cos\theta - x\sin\theta) - \alpha\sin\theta)\right| \leq T(L, \alpha, \beta, A, C) \\ \\ \left|y' - y - (-Cy + B - \beta x + \alpha y(y\cos\theta - x\sin\theta) + \alpha\cos\theta)\right| \leq T(L, \alpha, \beta, B, C) \end{array}\right.$$

with

$$\begin{array}{rl} T(L, \alpha, \beta, A, C) = & \Big[L^3\,\frac{2\alpha^2}{3} + L^2\left(\frac{4|C\alpha|}{3} + \frac{2|\beta\alpha|}{3} + \frac{4|\alpha|^3}{9}\right) \\ & +L\left(\alpha^2\left(2 + |\beta| + \frac{|C-1|}{3}\right) + \frac{4|A\alpha|}{3} + \frac{2|\beta C|}{3} + \frac{\beta^2}{3} + \frac{2C^2}{3} + \frac{|\beta|^3}{9}\right) \\ & +|\alpha|\left(\frac{2\beta^2}{3} + \frac{4|\beta|}{3} + \frac{4|C|}{3} + \frac{2|\alpha A|}{3} + \frac{8\alpha^2}{9}\right) + \frac{4|AC|}{3}\Big]. \end{array}$$

The proof of this theorem is given in Appendix B. Thanks to the parameter values given in table 1, the optical flow can be approximated by a quadratic formula in (x, y). Indeed, these parameter values allow to make the bound T small in comparison to the value of each component of optical flow. For example, in the case of a pure translation with $A = B = 0.09$ and $C = 0.03$, the bound T is equal to $4.2\ 10^{-3}$ for $L = 1$ and $8.4\ 10^{-3}$ for $L = 8$, whereas the components of optical flow have an order of magnitude of 10^{-2} or 10^{-1}. For a purely projective rotation with $\alpha = 0.01$, the optical flow has an order of 10^{-2} and the bound is equal to $3\ 10^{-4}$ for $L = 1$ and $5.2\ 10^{-3}$ for $L = 4$. For $L = 8$, the optical flow has an order of 10^{-1} and the bound is $3.6\,10^{-2}$.

If $L, \alpha, \beta, A, B, C$ are sufficiently small, the optical flow can be approximated by the sum of three independent terms; the component $(-Cx + A, -Cy + B)$ is due to the translation of the camera, $(\beta y, -\beta x)$ to the rotation R^k_β and $(\alpha\, x(-x\sin\theta + y\cos\theta) - \alpha\sin\theta, \alpha\, y\,(-x\sin\theta + y\cos\theta) + \alpha\cos\theta)$ to the rotation $R_{\theta,\alpha}$. These three terms are approximations of optical flows, respectively produced by the translation, the rotations R^k_β and $R_{\theta,\alpha}$.

Remarks

- Let us remark that at the image center, when x and y have 10^{-1} order (for a unit focal length), the quadratic term is negligible in comparison to the other terms. Thus, the deformation of the center of the image is mainly affine.

- At the beginning of this paper, we did assume that the translation t and the depth of the scene have to verify

$$\left(\frac{1}{Z_{inf}} - \frac{1}{Z_{sup}}\right)\|t\|\,\frac{2(L+1)}{3} \leq \varepsilon$$

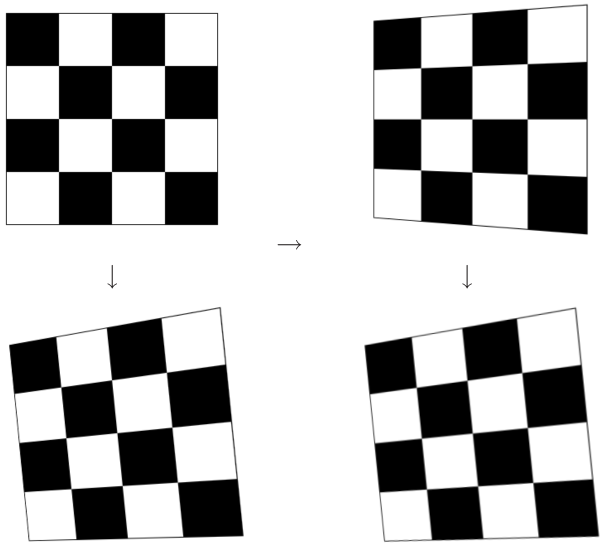

Figure 4: *Decomposition of deformation. On left, a checkerboard deformed by a camera motion. On right, the deformation can be decomposed in, first, a “purely” projective deformation, generated by the rotation $R_{\theta,\alpha}$ (at top) followed by a similarity (bottom).*

for substituting depths by a constant in formulas (1). As the approximation of optical flow has an order of 10^{-2}, we must choose an approximation error ε at least inferior to 10^{-2}.

3.5 Modelisation assets

In this section, we have first proposed to work in the registration group, well-adapted to the projective applications φ and ψ that link two consecutive images f and g. The advantage of this group is the isomorphism with the Lie group $SE(3)$, which allows to compose projective deformations through the composition of camera motions.

Second, we have described a new camera motion decomposition to emphasize two components of image deformation: a similarity and a “purely” projective deformation, due to the change of optical axis direction. This decomposition is interesting because it corresponds to a physical perception of camera motion effects on consecutive images. As shown on figure 4, we easily perceive the two deformations: the “purely” projective deformation, which deforms parallels on the checkerboard, and the similarity, which preserves angles. With this decomposition, we have obtained a quadratic approximation of optical flow for two consecutive images, where the quadratic term is only due to the change of optical axis direction. Remark that we only need condition (3) for approximating equation (1) by ψ.

4 Camera motion estimation

Let f and g be two adjacent images in a video sequence. In this section, we propose a method for estimating camera motion between f and g, based on camera motion decomposition and optical flow quadratic approximation.

4.1 Algorithm

Odobez and Bouthémy propose in [20] a method for determinating 2D parametric motions between two images. They use constant, affine or quadratic models. Their method is robust, multiresolution and only uses spatial and temporal gradients of intensity. The software, developped by the authors, is available at the address `http://www.irisa.fr/Vista/Motion2D`.

Let us now describe briefly their algorithm. The optical flow at a point (x, y) is assumed to be parametric, denoted $u_\Theta(x, y)$, where Θ is the set of parameters. Several models are proposed, the most general has 12 parameters

$$u_\Theta(x,y) = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} + \begin{pmatrix} a_1 & a_2 \\ a_3 & a_4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} q_1 & q_2 & q_3 \\ q_4 & q_5 & q_6 \end{pmatrix} \begin{pmatrix} x^2 \\ xy \\ y^2 \end{pmatrix}.$$

The displacement frame difference (DFD) associated to a parametric motion model at the point (x, y) is defined with

$$\mathrm{DFD}_{(\Theta,\xi)}(x,y) = g((x,y) + u(x,y)) - f(x,y) + \xi$$

where ξ is a global intensity shift to account for global illumination change. The set of parameters is thus estimated by minimizing the following function

$$\sum_{(x,y)\in f} \rho(\mathrm{DFD}_{(\Theta,\xi)}(x,y), \Gamma)$$

where the function ρ is called an M-estimator since its minimization corresponds to the maximum-likelihood estimation if ρ is considered as the opposite log-likelihood of the model. The authors choose a function bounded for high values in order to eliminate the contribution of outliers. They use the Tuckey's biweight function defined as

$$\rho(t,\Gamma) = \begin{cases} \frac{t^2}{2}(\Gamma^4 - \Gamma^2 t^2 + \frac{t^4}{3}) & \text{if } |t| < \Gamma, \\ \frac{\Gamma^6}{6} & \text{otherwise.} \end{cases}$$

The minimization of ρ is performed using an incremental and multiresolution scheme described in [20]. This method is accurate and has a low computational cost.

Several models are proposed in the software but none corresponds to our optical flow approximation. Thus, we have added the following model to the software

$$u_\Theta(x,y) = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} + \begin{pmatrix} a_1 & a_2 \\ -a_2 & a_1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} q_1 & q_2 & 0 \\ 0 & q_1 & q_2 \end{pmatrix} \begin{pmatrix} x^2 \\ xy \\ y^2 \end{pmatrix}.$$

Once the six parameters $(c_1, c_2, a_1, a_2, q_1, q_2)$ are estimated, we convert them into α, β, θ, A, B, C by identifying the previous expression with the quadratic formula given in theorem 2

$$\begin{cases} \theta = \begin{cases} -\arctan(q_1/q_2) & \text{if } q_2 > 0 \\ -\arctan(q_1/q_2) + \pi & \text{if } q_2 < 0 \\ \pi/2 & \text{if } q_2 = 0 \text{ and } q_1 > 0 \\ -\pi/2 & \text{if } q_2 = 0 \text{ and } q_1 \leq 0. \end{cases} \\ \alpha = \sqrt{q_1^2 + q_2^2} \\ \beta = a_2 \\ A = c_1 + \alpha \sin\theta \\ B = c_2 - \alpha\cos\theta \\ C = -a_1. \end{cases}$$

4.2 Results

The performances of our method are illustrated through camera motion estimations on synthetic and real sequences, and some applications of these estimations. The context for applicating our method is given by condition (3)

$$\left(\frac{1}{Z_{inf}} - \frac{1}{Z_{sup}}\right) \|t\| \, \frac{2(L+1)}{3} \leq \varepsilon,$$

with $\varepsilon < 10^{-2}$. This means that for a given image size, the product of translation norm and variations of inverse of depth must be small enough. We do not need condition (4) since we only use the deformation ψ.

4.2.1 Synthetic sequences

We first estimate camera motion on sequences, that we have created from an image, considered as orthogonal to the optical axis and deformed with sets of six parameters (θ, α, β, A, B, C). These sets are randomly generated with respect to values given in table 1. The angle of view is equal to $90°$. Three sequences of 200 images are synthesized; the first one is generated with translations, the second one with rotations and the third one with plain motions. The initial image is shown on figure 5. We assumed that depth is constant and apply formula (6) on the image with a bilinear interpolation.

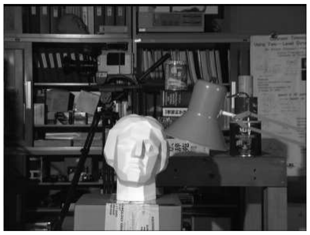

Figure 5: *Initial image for test sequences.*

	Translation direction error	Axis rotation direction error	Rotation angle error	
			absolute	relative
Plain motions	9.7°	17.3°	0.03°	2.2%
Pure translations	4.5°	-	0.01°	-
Pure rotations	-	18.2°	0.002°	0.1%

Table 2: *Results of camera motion estimations on 3 synthetic sequences of 200 images. The errors are averaged errors computed over each sequence.*

Camera motion results are shown on table 2. Whatever the type of camera motion, the estimations of translation direction are correct up to a few degrees and the estimated rotation direction up to ten or twenty degrees. These last errors may seem to be important but we must notice that the change of optical axis direction is hard to estimate, as small rotation and small translation can produce very similar results on images. For example, a small translation with direction i and a small rotation with axis j produce very close effects on images. The estimations of rotation angle are more accurate; they are correct up to a few hundredths degrees for rotation angles of 1 or 2 degrees. In sum, obtained results are rather good, better when motions are reduced to a translation or a rotation. Moreover, the scene was quite complicated and the method is very fast: it takes 7.7 seconds for a sequence of 200 images with 284×188 pixels, with a processor Pentium M 1.8 GHz.

Robustness Figure 6 shows the robustness of the algorithm to impulse or gaussian noise. We add various amounts of impulse or gaussian noise to the sequence produced with complete motions. Graphs plot errors in the estimates as a function of noise level, averaged over the 200 images at each noise level. For both types of noise, the errors do not increase a lot: they remain close to errors computed without noise, less than 15 degrees for translation direction, at most few tenths degrees for the angle of rotation (for impulse noise). Thus the method is robust, thanks to the use of M-estimator: it provides good results even when the amount of impulse noise is important.

Depths influence In this paper, we have approximated the deformation (equation (1)) between g and f by ψ, provided that condition (3) was verified, with $\varepsilon < 10^{-2}$

$$\left(\frac{1}{Z_{inf}} - \frac{1}{Z_{sup}}\right) \|t\| \, \frac{2(L+1)}{3} \leq \varepsilon.$$

The smaller is $\left(\frac{1}{Z_{inf}} - \frac{1}{Z_{sup}}\right) \|t\| \, \frac{2(L+1)}{3}$, the more accurate is the approximation. For a given scene, further the camera is from the scene, smaller is the previous expression and better is the estimation. This fact is illustrated with motion estimation on synthetic sequences SOFA5 and SOFA6 (Sequences for Optical Flow Analysis, courtesy of the Computer Vision Group, Heriot-Watt University). Each sequence, which each contains 20 images, is given with internal and external camera parameters, and camera motion. Motions are basic: a translation of direction k for SOFA5 and a rotation with axis k followed by a translation with direction k for SOFA6. Images of the two sequences are shown on figure 7. Results are given on tables 4 and 5; the evaluation of $\left(\frac{1}{Z_{inf}} - \frac{1}{Z_{sup}}\right) \|t\| \, \frac{2(L+1)}{3}$ is also computed (in units of focal length) on table 3.

	$\frac{1}{Z_{inf}} - \frac{1}{Z_{sup}}$	$\left(\frac{1}{Z_{inf}} - \frac{1}{Z_{sup}}\right) \|t\| \, \frac{2(L+1)}{3}$
Image 1	0.0062	0.0076
Image 10	0.0112	0.0137
Image 20	0.0293	0.0357

Table 3: *Relative variations of inverse of depths in sequences SOFA5 et SOFA6. Depths* Z_{inf} *and* Z_{sup}, $\|t\|$ *and* L *are expressed in units of focal length in the camera system.*

As the camera comes close the scene, differences in table 3 increase in time. Remark that we have $L \leq 8$; the angle of view is equal to $45°$. Tables 4 and 5 give errors in motion estimation between consecutive images at three instants: at the beginning of the sequence, at the middle and at the end. The estimation method is the same as previously used: we assume no *a priori* type of motion. For SOFA5, the translation direction estimates are very good, better than on previous synthetic sequences. This is due to the motion simplicity and to the fixity of optical axis. However, we observe that when the camera comes close the scene, the translation estimation error and the rotation angle estimation (that should be null)

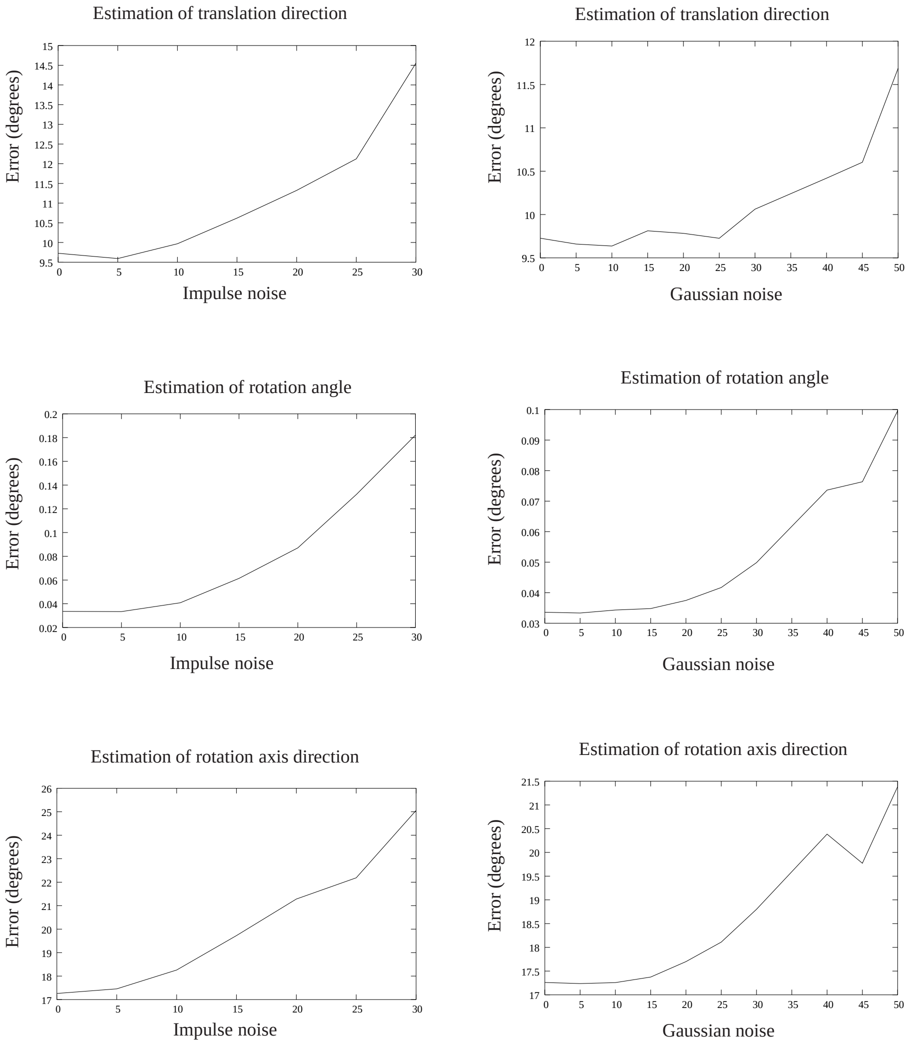

Figure 6: *Camera motion estimation errors, averaged over 200 images of the noisy sequence. Impulse noise level of 10 means that* 10% *of pixels values are randomly chosen with a uniform variable distributed on all gray levels. Gaussian noise level of* 10 *means that we add to the images a gaussian noise with standard deviation 10.*

slightly increase. For SOFA6, the translation direction estimates are always very good; but the estimation errors on axis and angle of rotation increase significantly when the camera comes close the scene.

Although errors increase when we get close to the scene (because we then are away from the defined context), our method allows to conclude for simple motions (for example when the optical axis is fixed) even if condition (3) is not verified with $\varepsilon < 10^{-2}$.

4.2.2 Applications on real sequences

As we have no real sequences with given camera motion and internal camera parameters, we illustrate the quality of camera motion estimation with two applications of estimation results.

The first use is mosaicing. In our framework, we suppose that two successive images are linked by a planar transformation, thus the knowledge of camera motion between these

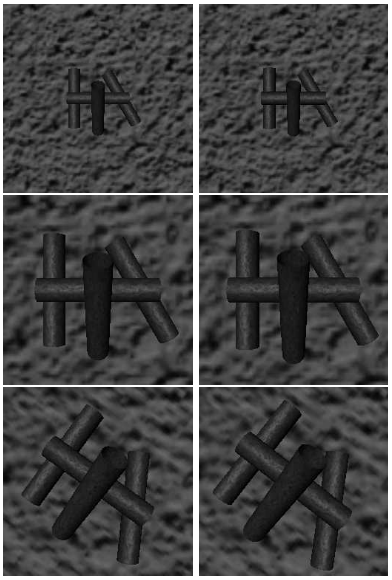

Figure 7: *At the top, images 1 and 2 of SOFA5 and SOFA6. At the middle, images 19 and 20 of SOFA5 and at the bottom, images 19 and 20 of SOFA6.*

	Translation direction error	Rotation angle error
Between images 1 and 2	0.12°	0.0005°
Between images 10 and 11	0.17°	0.0018°
Between images 19 and 20	0.55°	0.019°
Errors average	0.42°	0.014°

Table 4: *Estimation errors on SOFA5. Camera motion is constant on the sequence: it is a translation of direction k (the camera comes close the scene).*

two images allows to register one image to the other. With the estimation of camera motion on a whole sequence, we can compute the motion between two images distant in time, by composing displacement estimations in the registration group. Thus, by choosing an image viewpoint and registering some images distant in time on it, we obtain a bigger image that we could observe from the image viewpoint, but with a larger vision field. Figures 8 and 9 show two panoramas, computed with the estimated camera motion on a real video sequence of an office. Remark that the mosaicing is theoretically possible if the viewpoint does not change (when there is no translation) or when the camera films a planar scene. Our movie does not exactly verify the hypothesis of pure rotation because although the camera translation is very small between adjacent frames, it may be significant between two images distant in time and obviously, the scene is not planar. But as the scene is rather far from the camera location, registrations are correct.

The second use is augmented reality. It consists in adding an object in a sequence in such a way it appears to be present in the scene. In our framework, the application is

	Translation direction error	Rotation axis direction error	Rotation angle error	
			absolute	relative
Between images 1 and 2	0.23°	0.001°	0.051°	2.5%
Between images 10 and 11	0.38°	0.491°	0.068°	3.4%
Between images 19 and 20	0.97°	1.08°	0.094°	4.7%
Errors average	0.39°	0.269°	0.069°	3.4%

Table 5: *Estimation errors on SOFA6. Camera motion is constant on the sequence: it is a rotation of axis k followed by a translation of direction k (the camera comes close the scene).*

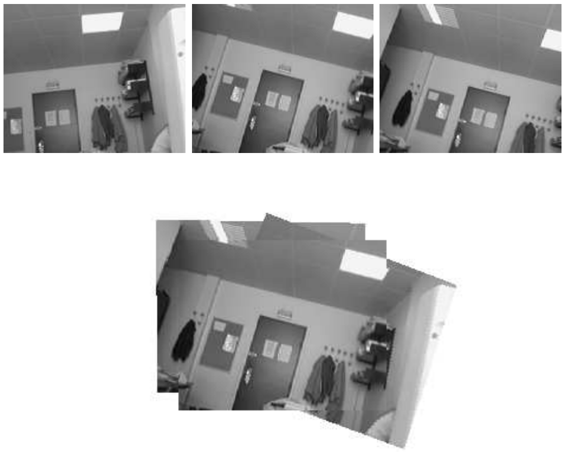

Figure 8: *At the top, scenes 20, 35 and 50 of the office sequence; at the bottom, reconstructed panoramic view on viewpoint 35.*

simplified since we insert in the office sequence a planar object, which is a poster. This poster is first inserted on the main planar region of the scene, roughly parallel to the retinal plane. Next, it is deformed with the projective application 6 associated to the estimated camera motion. Example frames from the augmented sequence are presented on figure 10. This experience shows that the camera motion is accurately estimated: the poster moves with the same motion as the background of the scene. More precisely, the poster orientation follows the orientation of the background (camera rotations are correctly estimated) and its position is plausible.

Let us recall that our goal is not mosaicing nor augmented reality: these two applications are utilizations of estimated camera motions and illustrate the quality of our motion estimation results in our framework.

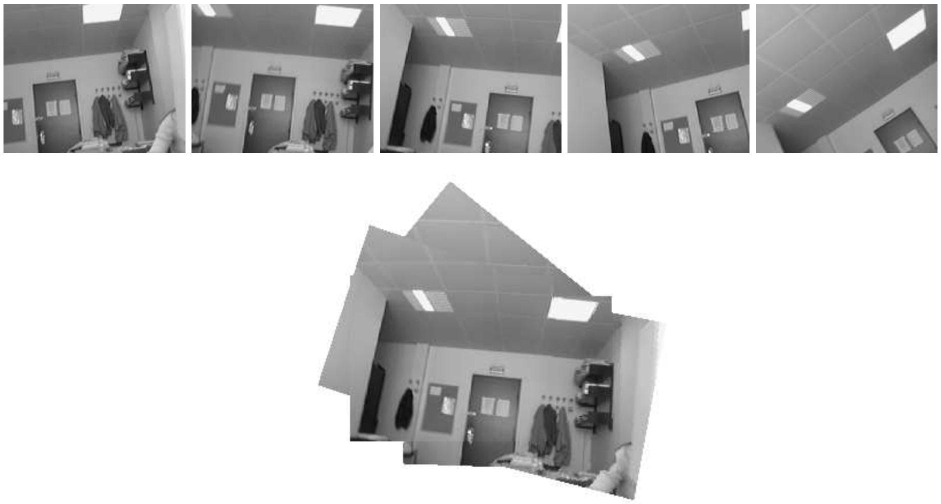

Figure 9: *At the top, scenes 10, 30, 60, 70 and 80 of the office sequence; at the bottom, reconstructed panoramic view on viewpoint 60.*

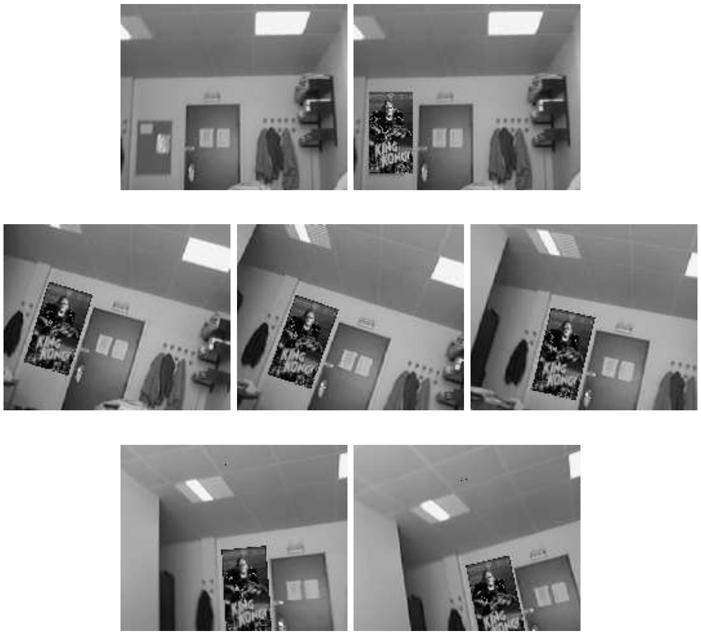

Figure 10: *Replacement of the notice board by a cinema poster. At the top: the insertion of the poster on the first image. At the middle, images* $10, 20, 30, 40$ *et* 45 *of the new sequence obtained by deforming the poster with the estimations of camera motions and pasting it in the sequence.*

5 Conclusion

In this paper, we have proposed a new global method for the problem of egomotion estimation, well-adapted to adjacent frames as produced by a camera that films a static scene, when variations of inverse of scene depths and translation are sufficiently small. This context is theoretically limited, but as the translation is very small between two acquisitions, it is not so restrictive. In this context, the method is very fast : first because we do not have to compute optical flow or match points as it is a direct method, second because of the multiresolution scheme in the software Motion2D, fitted to our quadratic approximation of optical flow. It is also robust, thanks to the use of an M-estimator. Moreover, the modeling of camera motion in the registration group allows to compose image deformations and to obtain camera motion between two images distant in time in a sequence. At last, as it is a global method, it is robust to a moving object in the scene, provided its size is limited in comparison to the image size.

A Proof of theorem 1

Let $0 < Z_{inf} \leq Z_0 \leq Z_{sup}$ and (x, y) belong to K. We denote $\delta = \frac{1}{Z(x,y)} - \frac{1}{Z_0}$. Thus, we can write formula 1

$$\left\{\begin{array}{l} x' = \dfrac{u_0^1 - \delta\langle t, R(i)\rangle}{v_0 - \delta\langle t, R(k)\rangle} \\ \\ y' = \dfrac{u_0^2 - \delta\langle t, R(j)\rangle}{v_0 - \delta\langle t, R(k)\rangle} \end{array}\right.$$

where

$$\left\{\begin{array}{l} u_0^1 = a_1 x + a_2 y + a_3 - \langle \frac{t}{Z_0}, R(i)\rangle \\ u_0^2 = b_1 x + b_2 y + b_3 - \langle \frac{t}{Z_0}, R(j)\rangle \\ v_0 = c_1 x + c_2 y + c_3 - \langle \frac{t}{Z_0}, R(k)\rangle. \end{array}\right.$$

By applying Taylor's formula on δ about 0 with integral form of remainder, we obtain

$$\left\{\begin{array}{l} x' = \dfrac{u_0^1}{v_0} + \displaystyle\int_0^\delta \frac{\langle t, R(k)\rangle\, u_0^1 - \langle t, R(i)\rangle\, v_0}{\left(v_0 - z\,\langle t, R(k)\rangle\right)^2}\, dz = \frac{u_0^1}{v_0} + \delta\, \frac{\langle t, R(k)\rangle\, u_0^1 - \langle t, R(i)\rangle\, v_0}{v_0\,(v_0 - \delta\,\langle t, R(k)\rangle)} \\ \\ y' = \dfrac{u_0^2}{v_0} + \displaystyle\int_0^\delta \frac{\langle t, R(k)\rangle\, u_0^2 - \langle t, R(j)\rangle\, v_0}{\left(v_0 - z\,\langle t, R(k)\rangle\right)^2}\, dz = \frac{u_0^2}{v_0} + \delta\, \frac{\langle t, R(k)\rangle\, u_0^2 - \langle t, R(j)\rangle\, v_0}{v_0\,(v_0 - \delta\,\langle t, R(k)\rangle)} \end{array}\right.$$

that implies

$$\left\{\begin{array}{l} \left|\dfrac{\langle t, R(k)\rangle\, u_0^1 - \langle t, R(i)\rangle\, v_0}{v_0\,(v_0 - \delta\,\langle t, R(k)\rangle)}\right| \leq \|t\|\, \dfrac{|u_0^1| + |v_0|}{|v_0|}\, \left|\dfrac{1}{v_0 - \delta\,\langle t, R(k)\rangle}\right| \\ \\ \left|\dfrac{\langle t, R(k)\rangle\, u_0^2 - \langle t, R(j)\rangle\, v_0}{v_0\,(v_0 - \delta\,\langle t, R(k)\rangle)}\right| \leq \|t\|\, \dfrac{|u_0^2| + |v_0|}{|v_0|}\, \left|\dfrac{1}{v_0 - \delta\,\langle t, R(k)\rangle}\right|. \end{array}\right.$$

Since $(x, y) \in K \subseteq [-\frac{L}{2}, \frac{L}{2}]^2$, we have, with the hypothesis 2

$$\left\{\begin{array}{l} \dfrac{|u_0^1| + |v_0|}{|v_0|} \leq \left|\dfrac{u_0^1}{v_0} - x\right| + |x| + 1 \leq L + 1 \\ \\ \dfrac{|u_0^2| + |v_0|}{|v_0|} \leq \left|\dfrac{u_0^2}{v_0} - y\right| + |y| + 1 \leq L + 1. \end{array}\right.$$

Moreover, as the hypothesis 1 implies

$$\left|\frac{1}{v_0 - \delta\,\langle t, R(k)\rangle}\right| \leq \frac{4}{3},$$

thus

$$\max\left(\left|x' - \frac{u_0^1}{v_0}\right|, \left|y' - \frac{u_0^2}{v_0}\right|\right) \leq \delta\,\|t\|\,\frac{4(L+1)}{3}.$$

Now, if

$$\left(\frac{1}{Z_{inf}} - \frac{1}{Z_{sup}}\right) \|t\| \, \frac{2(L+1)}{3} \leq \varepsilon,$$

then, for Z_0 such that $\frac{1}{Z_0} = \frac{1}{2}\left(\frac{1}{Z_{inf}} + \frac{1}{Z_{sup}}\right)$, we have

$$\forall (x,y) \in K, \quad \left|\frac{1}{Z(x,y)} - \frac{1}{Z_0}\right| \|t\| \, \frac{4(L+1)}{3} \leq \varepsilon,$$

that implies

$$\forall (x,y) \in K, \quad \max\left(\left|x' - \frac{u_0^1}{v_0}\right|, \left|y' - \frac{u_0^2}{v_0}\right|\right) \leq \varepsilon.$$

B Proof of theorem 2

Let $D = (\theta, \alpha, \beta, A, B, C)$ be a camera motion. The rotation matrix R is equal to

$$\begin{pmatrix} \cos\beta - (1-\cos\alpha)\sin\theta\sin(\theta-\beta) & -\sin\beta + (1-\cos\alpha)\sin\theta\cos(\theta-\beta) & \sin\theta\sin\alpha \\ \sin\beta + (1-\cos\alpha)\cos\theta\sin(\theta-\beta) & \cos\beta - (1-\cos\alpha)\cos\theta\cos(\theta-\beta) & -\cos\theta\sin\alpha \\ -\sin\alpha\sin(\theta-\beta) & \sin\alpha\cos(\theta-\beta) & \cos\alpha \end{pmatrix}$$

that we also denote

$$R = \begin{pmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{pmatrix}.$$

The coefficients of R verify, by using Taylor expansions in α and β

$$\left\{ \begin{array}{lll} a_1 = 1 + k_{a_1}, & k_{a_1} = o(\beta) + o(\alpha), & |k_{a_1}| \leq \beta^2/2 + \alpha^2/2(1+|\beta|) \\ a_2 = \beta + k_{a_2}, & k_{a_2} = o(\beta^2) + o(\alpha), & |k_{a_2}| \leq \beta^3/6 + \alpha^2/2(1+|\beta|) \\ a_3 = -\alpha\sin\theta + k_{a_3}, & k_{a_3} = o(\alpha^2) + o(\sqrt{|\alpha\beta|}), & |k_{a_3}| \leq \alpha^3/6 + |\alpha|(|\beta| + \beta^2/2) \\ b_1 = -\beta + k_{b_1}, & k_{b_1} = o(\beta^2) + o(\alpha), & |k_{b_1}| \leq \beta^3/6 + \alpha^2/2(1+|\beta|) \\ b_2 = 1 + k_{b_2}, & k_{b_2} = o(\beta) + o(\alpha), & |k_{b_2}| \leq \beta^2/2 + \alpha^2/2(1+|\beta|) \\ b_3 = \alpha\cos\theta + k_{b_3}, & k_{b_3} = o(\alpha^2) + o(\sqrt{|\alpha\beta|}), & |k_{b_3}| \leq \alpha^3/6 + |\alpha|(|\beta| + \beta^2/2) \\ c_1 = \alpha\sin\theta + k_{c_1} & k_{c_1} = o(\alpha^2), & |k_{c_1}| \leq |\alpha|^3/6 \\ c_2 = -\alpha\cos\theta + k_{c_2} & k_{c_2} = o(\alpha^2), & |k_{c_2}| \leq |\alpha|^3/6 \\ c_3 = 1 + k_{c_3} & k_{c_3} = o(\alpha), & |k_{c_3}| \leq |\alpha|^2/2. \end{array} \right.$$

According to the definition of the application ψ, we have

$$\left\{ \begin{array}{l} x' - x = \dfrac{x + \beta y - \alpha\sin\theta + A + o(\alpha) + o(\beta) + o(\sqrt{|\alpha\beta|})}{\alpha\sin\theta\, x - \alpha\cos\theta\, y + 1 + C + o(\alpha)} - x \\[2em] y' - y = \dfrac{y - \beta x + \alpha\cos\theta + B + o(\alpha) + o(\beta) + o(\sqrt{|\alpha\beta|})}{\alpha\sin\theta\, x - \alpha\cos\theta\, y + 1 + C + o(\alpha)} - y, \end{array} \right.$$

that is

$$\left\{ \begin{array}{ll} x' - x = & \Big(x + \beta y - \alpha\sin\theta + A + o(\alpha) + o(\beta) + o(\sqrt{|\alpha\beta|})\Big) \\ & (1 - C - \alpha\sin\theta\, x + \alpha\cos\theta\, y + o(\alpha) + o(C)) - x \\[1em] y' - y = & \Big(y - \beta x + \alpha\cos\theta + B + o(\alpha) + o(\beta) + o(\sqrt{|\alpha\beta|})\Big) \\ & (1 - C - \alpha\sin\theta\, x + \alpha\cos\theta\, y + o(\alpha) + o(C)) - y. \end{array} \right.$$

That implies

$$\left\{\begin{array}{ll} x'-x= & -Cx+\beta y-\alpha\sin\theta+A-\alpha\sin\theta\, x^2+\alpha\cos\theta\, xy+o(\alpha)+o(\beta)+o(C) \\ & +o(\sqrt{|\alpha\beta|})+o(\sqrt{|C\beta|})+o(\sqrt{|C\alpha|})+o(\sqrt{|\alpha A|})+o(\sqrt{|CA|}) \\ & \\ y'-y= & -Cy-\beta x+\alpha\cos\theta+B-\alpha\sin\theta\, xy+\alpha\cos\theta\, y^2+o(\alpha)+o(\beta)+o(C) \\ & +o(\sqrt{|\alpha\beta|})+o(\sqrt{|C\beta|})+o(\sqrt{|C\alpha|})+o(\sqrt{|\alpha B|})+o(\sqrt{|CB|}). \end{array}\right.$$

Furthermore,

$$\begin{array}{l} \big|x'-x-\big(-Cx+\beta y-\alpha\sin\theta+A-\alpha\sin\theta\, x^2+\alpha\cos\theta\, xy\big)\,\big| \\ \\ =\left|\frac{-c_1x^2-c_2xy+(a_1-c_3-C)x+a_2y+a_3+A-(c_1x+c_2y+c_3+C)(A-Cx+\beta y+\alpha\cos\theta xy-\alpha\sin\theta x^2-\alpha\sin\theta)}{c_1x+c_2y+c_3+C}\right|. \end{array}$$

By using bounds of $|k_{a_1}|, |k_{a_2}|, \ldots, |k_{c_3}|$ and the hypothesis 1, we get

$$\begin{array}{ll} \multicolumn{2}{l}{\big|x'-x-\big(-Cx+\beta y-\alpha\sin\theta+A-\alpha\sin\theta\, x^2+\alpha\cos\theta\, xy\big)\,\big|} \\ \leq \frac{4}{3} & \big|\,x^2(-c_1+Cc_1+\alpha\sin\theta c_3+\alpha\sin\theta C)-y^2\beta c_2+ \\ & xy(-c_2+Cc_2-\beta c_1-\alpha\cos\theta c_3-\alpha\cos\theta C)+x^2y(-c_1\alpha\cos\theta+c_2\alpha\sin\theta)+ \\ & x^3(\alpha\sin\theta c_1)-xy^2c_2\alpha\cos\theta+x(a_1-c_3-C-Ac_1+c_1\alpha\sin\theta+Cc_3+C^2)+ \\ & y(a_2-Ac_2+c_2\alpha\sin\theta-\beta c_3-\beta C)+a_3+A(1-c_3-C)+\alpha\sin\theta(c_3+C)\big|. \end{array}$$

As $(x,y)\in[-L/2,L/2]^2$, we obtain

$$\begin{array}{ll} \multicolumn{2}{l}{\big|x'-x-\big(-Cx+\beta y-\alpha\sin\theta+A-\alpha\sin\theta\, x^2+\alpha\cos\theta\, xy\big)\,\big|} \\ \\ \leq & \Big[L^3\,\frac{2\alpha^2}{3}+L^2\left(\frac{4|C\alpha|}{3}+\frac{2|\beta\alpha|}{3}+\frac{4|\alpha|^3}{9}\right) \\ \\ & +L\left(\alpha^2\left(2+|\beta|+\frac{|C-1|}{3}\right)+\frac{4|A\alpha|}{3}+\frac{2|\beta C|}{3}+\frac{\beta^2}{3}+\frac{2C^2}{3}+\frac{|\beta|^3}{9}\right) \\ \\ & +|\alpha|\left(\frac{2\beta^2}{3}+\frac{4|\beta|}{3}+\frac{4|C|}{3}+\frac{2|\alpha A|}{3}+\frac{8\alpha^2}{9}\right)+\frac{4|AC|}{3}\Big]. \end{array}$$

By a similar way, we bound $\big|y'-y-\big(-Cy-\beta x+\alpha\cos\theta+B-\alpha\sin\theta\, xy+\alpha\cos\theta\, y^2\big)\,\big|$ by replacing A with B.

References

[1] "A. Azarbayejani, A. P. Pentland, Recursive estimation of motion, structure and focal length", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 17(6), pp. 562-575, 1995.

[2] A. Yao, A. Calway, "Robust estimation of 3-d camera motion for uncalibrated augmented reality", Dept of Computer Science, University of Bristol, CSTR-02-001, 2002.

[3] H.C. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two projections", *Nature*, Vol. 293(10), pp. 133-135, 1981.

[4] O. Faugeras, *Three-Dimensional Computer Vision, a geometric Viewpoint*, MIT Press, 1993.

[5] O. Faugeras, S. Maybank, "Motion from point matches: multiplicity of solutions", *International Journal of Computer Vision*, Vol. 4(3), pp. 225-246, 1990.

[6] T. Huang, O. Faugeras, “Some properties of the Ematrix in two-view motion estimation”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 11(12), pp. 1310-12, 1989.

[7] O. Faugeras, Q.T. Luong, T. Papadopoulo, *The Geometry of Multiple Images*, MIT Press, 2000.

[8] A.R. Bruss, B.K. Horn, “Passive navigation”, *Computer Graphics and Image Processing*, Vol. 21, pp. 3-20, 1983.

[9] D. Heeger, A. Jepson, “Subspace Methods for Recovering Rigid Motion I: Algorithm and Implementation”, *International Journal of Computer Vision* , Vol. 7(2), pp. 95-117, 1992.

[10] Y. Ma, J. Koseckà, S. Sastry, “Linear Differential Algorithm for Motion Recovery: A Geometric Approach”, *International Journal of Computer Vision*, Vol. 36(1), pp. 71-89, 2000.

[11] M. J. Brooks, W. Chojnacki, L. Baumela, “Determining the ego-motion of an uncalibrated camera from instantaneous optical flow”, *Journal of the Optical Society of America*, Vol. A 14(10), pp. 2670-2677, 1997.

[12] C. Tomasi, J. Shi, “Direction of heading from image deformations”, in *IEEE Conf. on Computer Vision and Pattern Recognition*, 1993, pp. 422-427.

[13] J. Lawn, R. Cipolla, “Robust Egomotion Estimation from Affine Motion Parallax”, in *Proc. 3rd European Conf on Computer Vision*, Stockholm, Sweden, 1994, pp. 205-210.

[14] T.Y. Tian, C. Tomasi, D.J. Heeger, ”Comparison of Approaches to Egomotion Computation”, in *Proc. of Conf. on Computer Vision and Pattern Recognition*, 1996, pp. 315-320.

[15] M. Irani, B. Rousso, S. Peleg, “Recovery of Ego-motion using Region Alignement”, in *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 19(3), pp. 268-272, 1997.

[16] B.K. Horn, E.J. Weldon, “Direct Methods for Recovering Motion”, *International Journal of Computer Vision*, Vol. 2, pp.51-76, 1988.

[17] J.R. Bergen, P. Anandan, K.J. Hanna, R. Hingorani, “Hierarchical Model-Based Motion Estimation”, in *Proc. of European Conf. on Computer Vision and Pattern Recognition*, 1992, Vol. 2, pp. 237-252.

[18] S. Negahdaripour, B.K.P. Horn, “Direct passive navigation”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 9(1), pp.168-176, 1987.

[19] F. Dibos, G. Koepfler, P. Monasse, “Image Alignment”, *Geometric Level Set Methods in Imaging, Vision and Graphics*, Springer, 2003.

[20] J.M. Odobez, P. Bouthemy, “Robust Multiresolution Estimation of Parametric Motion Models”, *Jal. of Visual Communication and Image Representation*, Vol. 6(4), pp. 348-365, 1995.